



Big Fish eCommerce

BF Address Verification.doc

Last Updated: 15-Jan-2013

TABLE OF CONTENTS

<u>1</u>	<u>Modification History</u>	3
<u>2</u>	<u>Overview</u>	4
2.1	<u>General</u>	4
2.2	<u>Responsive Services</u>	4
2.3	<u>Verification Services</u>	4
2.4	<u>Scope</u>	5
2.5	<u>System Parameters</u>	5
<u>3</u>	<u>Responsive Processing</u>	6
3.1	<u>Visual Guideline</u>	6
3.2	<u>Processing Rules</u>	6
<u>4</u>	<u>Verification Processing</u>	7
4.1	<u>Visual Guideline</u>	7
4.2	<u>Processing Rules</u>	7
4.3	<u>Address is incorrect</u>	7
4.4	<u>Address is Incorrect, Alternatives Available</u>	8

1 Modification History

Date	Who	Comments
3-Oct-12	Salmon	Initial
15-Jan-13	Salmon	Clarifications based on dev feedback. Renamed from BF-Address-Validation

2 Overview

2.1 General

- This document will describe the verification for all addresses
- Standard validation should be applied as defined in the spec "*BF Customer Account Management.doc*"
 - In other words, the "verification" process described in this document assumes that standard "validation" has already taken place and the basic entry has been made by the customer
- External services
 - May be used to obtain accurate addresses, or be used for verification
 - "Responsive" services will provide for easy lookup of addresses based on key information (example: PostCodeAnywhere)
 - "Verification" services will allow for customer entry of the address and then be used for verification purposes (example: Melissa Data)

2.2 Responsive Services

- PostCodeAnywhere:
 - This implementation is "responsive" in that it acts less as a verification component and more as a helper, often with "auto complete" features
 - Customers are prompted for key address information in a drop-down entry box
 - Matching results are then displayed
 - The Customer can pick from one of the available matches
 - BigFish will complete the Address entry (City, State etc.)

2.3 Verification Services

- Melissa Data
 - This service is a typical "verification" process
 - The Customer will enter all relevant address information
 - This information is then verified via a service call
 - Errors are highlighted as appropriate
 - Resources:
 - <http://www.melissadata.com/free-trials/data-quality-suite.htm>
 - <http://www.melissadata.com/download.htm>
 - <http://www.melissadata.com/tech/addressobject.htm>
 - <http://www.melissadata.com/tech/webSMART.htm>
- US Post Office
 - Future implementation
- UPS (United Parcel Service)
 - Resources:
 - https://www.ups.com/upsdeveloperkit/downloadresource?loc=en_US

- Additional Processing
 - In all cases the Customer must have the ability to “use entered address” regardless of verification errors
 - Depending on the service an additional feature may be “did you mean” with a presentation of possible correct addresses

2.4 Scope

- Address Verification should be applied wherever addresses can be entered or edited, including:
 - Registration
 - Address Book
 - Check Out
 - Request Catalog

NOTE: It is possible that a client implementation will have both a “responsive” and a “validation” component

2.5 System Parameters

- A single system parameter ADDRESS_VERIFICATION_METHOD will be used to define what type of verification, if any, will be active in a specific implementation
- Current values are:
 - NONE
 - PCA
 - **TODO: this will replace PCA_ACTIVE_FLAG**
 - MELISSA_DATA
- PCA (PostCodeAnywhere parameters)
 - See spec “*BF Admin Module Admin.doc*”
- Melissa Data parameters:

MELISSA_LICENSE	A current license string is required to use the functionality of the Melissa address tool. Use 'DEMO' string as license in demo mode.
MELISSA_FILE_PATH_US	Indicates the Directory location where the Melissa US db is contained for checking the US address.
MELISSA_FILE_PATH_CAN	Indicates the Directory location where the Melissa Canada db is contained for checking the Canadian address.
MELISSA_VERIFICATION_MODE	Melissa provides the two types of services for address verification: first use the Melissa address tool locally and second send HTTP request to Melissa. Possible values are HTTP or FILEPATH.
MELISSA_HTTP_URL	HTTP url for address verification. Default value is https://addresscheck.melissadata.net/v2/REST/Service.svc/do
MELISSA_REGISTRATION_ID	Registered customer Id for web data verification

3 Responsive Processing

3.1 Visual Guideline

Address	
Responsive Lookup:	<input type="text" value="Enter partial address here and I will find a match for you"/>
Address 1:	<input type="text"/>
Address 2:	<input type="text"/>
City:	<input type="text"/>
State:	<input type="text" value="Drop down"/>
Zip:	<input type="text"/>
<input type="button" value="Submit"/>	

3.2 Processing Rules

- The responsive lookup component will attempt to match to known addresses
- Dynamic results may be displayed in a drop-down or popup component
- The customer selects on a result and the Address fields on the form are auto-populated from that selection

4 Verification Processing

4.1 Visual Guideline

Address	
Country:	<input type="text" value="Drop-down"/>
Address 1:	<input type="text"/>
Address 2:	<input type="text"/>
City:	<input type="text"/>
State:	<input type="text" value="Drop down"/>
Zip:	<input type="text"/>
<input type="button" value="Submit"/>	

4.2 Processing Rules

- Depending on the configuration, a verification service is executed
- There are 3 results that should be handled:
 - Address is OK, no further processing required
 - Address is incorrect, customer has the option of correcting or stating that the address as entered should be used
 - Address is incorrect, and the service call has identified alternatives. Customer also has the option to use the entered address

4.3 Address is incorrect

[Your Address is incorrect, please verify. If the address is correct, then check the "Use As Entered" checkbox and continue.]

Address	
Country:	<input type="text" value="United States"/>
Address 1:	<input type="text" value="1 Main Street"/>
Address 2:	<input type="text"/>
City:	<input type="text" value="Mainville"/>
State:	<input type="text" value="New York"/>

Zip: <input type="text" value="11223"/> <input type="button" value="Submit"/>
<input type="checkbox"/> Address as entered above is correct

- A checkbox is displayed that will allow for the customer to use the Address as entered (or modified)
- If the checkbox is checked, then the Submit action will accept the entered address. This is the case even if the customer modifies the information on the form.
- If the customer is NOT checked, then the verification service will be executed once more. Processing is then dependent on the validity of the entered address.
- The checkbox must be part of the DIV-Sequencing strategy so that it can be placed above or below the entered address

4.4 **Address is Incorrect, Alternatives Available**

- **NOTE:** we do not currently have a client requirement for this processing flow. This should be considered but **NOT IMPLEMENTED**

[Your Address is incorrect, please verify. If the address is correct, then check the "Use As Entered" checkbox and continue.]

Address	
Country:	<input type="text" value="United States"/>
Address 1:	<input type="text" value="1 Main Street"/>
Address 2:	<input type="text"/>
City:	<input type="text" value="Mainville"/>
State:	<input type="text" value="New York"/>
Zip:	<input type="text" value="11223"/>
<input type="checkbox"/> Address as entered above is correct	
<input type="button" value="Submit"/>	
Did You Mean ...	
Country:	United States
Address 1:	11 Main Avenue
City:	Mineola
State:	New York
Zip:	11501
<input type="button" value="Use This Address"/>	

- The button "Use This Address" will paste the information from the Alternate Address section into the Main Address section
- A checkbox is displayed that will allow for the customer to use the Address as entered (or modified)
- If the checkbox is checked, then the Submit action will accept the entered address. This is the case even if the customer modifies the information on the form.
- If the customer is NOT checked, then the verification service will be executed once more. Processing is then dependent on the validity of the entered address.
- The checkbox and alternate address must be part of the DIV-Sequencing strategy so that it can be placed above or below the entered address
- Melissa Data Implementation Notes
 - Melissa Data typically returns a "corrected" address
 - For example, an entered value of "1 Mai St" will be corrected as "1 Main Street"
 - The implementation will need to compare the "entered" data with the "corrected" data in order to comply with the UI guideline above (only if the Melissa Data service has modified the address)
- Future Implementation Notes
 - Some implementations may return a list of possible addresses
 - For example, an entered value of "29 Corfton" may return "29 Corfton Road Flat 1", and "29 Corfton Road, Flat 2" etc.
 - This specification will require clarification when and if this situation occurs